Neural Modeling of Face Animation for Telecommuting in Virtual Reality

THOMAS P. CAUDELL, ADAM L. JANIN, AND SEAN K. JOHNSON[†]
Boeing Computer Services
MS 7L-66
P.O. Box 24346
Seattle, WA 98124-0346

Abstract

Neural networks are used to generate facial animation models for use in virtual reality telecommuting systems. 2D face silhouettes are used to train and test multilayer perceptrons with backpropagation learning. This approach overcomes the problems encountered with the integration of face sensing devices and visual displays.

1. Introduction

The term "Telecommuting" was originated by Jack Nilles in 1975 [1] to describe the concept of people working together in remote physical locations using personal computers and telephones. The socioeconomical advantages and disadvantages of telecommuting have been studied by many researchers ([2], [3]). The prime advantages are higher quality work output, lower worker stress, longer uninterrupted work sessions, reduced ware on the transportation infrastructure, reduced use of petrochemicals, and improved worker job satisfaction. The prime disadvantages are management fear of control loss, lack of personal face-to-face interactions leading to a feeling of isolation by telecommuting workers and supervisors, and the basic cost of telecommuting hard- and software. Although several companies have started telecommuting programs on a voluntary bases for jobs that involve information processing or management, such as secretarial and computer programmers, the practice is not widely used in industry. The perceived need for face-to-face workplace interactions verses the trend towards more "cocooning" in the home [4], has lead to a dilemma in the effective use of the telecommuting paradigm.

Immersive virtual reality (VR) provides a technological solution to the "face-to-face/cocooning" dilemma in telecommuting. The idea is to equip the telecommuter with a VR system that is connected through the telephone system to other telecommuters with similar equipment. With head-mounted graphical displays, body-suit interaction systems, 3D sound systems and a microphone, the worker will perceive and interact with coworkers in a natural way, as if they are in each other's physical presence. The coordinates of the person's body components, such as his head, arms, torso, and legs, as well as audio signals would be transmitted to the receiving station, where a graphics engine animates a polygonal human model of the transmitter. The receiver views this in his local head-mounted display. With suitable compression techniques, the signal bandwidth for near real time communication is well within that found in standard telephone transmission lines. Several people could simultaneously work together within this virtual workplace, forming a strong team spirit, while never leaving home.

The dilemma is not solved completely by the above scenario. A critical missing component is the transmission of facial information. Although "body language" conveys much, the face is the primary conveyor of personal feelings and attitudes. Face transmission poses a practical problem in VR telecommuting systems -- unobstructed view of the users face. If not for the fact that part of the face is covered with a video display system, a standard video camera could image the face in motion, compress the video, transmit it over the telephone lines to the receiver's graphics engine, which could superimpose it on the human model. Unfortunately, this is not possible with the current generation of VR head mounted displays.

[†] Morehouse College, Atlanta, GA

0-7803-1363-1/93 $3.00 © 1993 IEEE

The solution known to several researchers ([5], [6], [7], [8]) is to model the dynamics of the face in motion using either interpolation techniques or anatomical face muscle models. With such a model, a reduced number of parameters of the model needs to be transmitted to the receiving station for animation. Colors and textures can be added to the face models to add realism. Real time models based on facial anatomy have been developed in the last few years that execute on graphics workstations [9]. Other researchers have developed a system that tracks special features on the face from which they control texture maps for the animation of face models [10]. There are three primary disadvantages of these approaches: 1) it is difficult to rapidly customize the model for each individual telecommuter's face, 2) one must develop methods to sense facial features under the graphics video display, and 3) a great deal of processing power is necessary to execute anatomical models.

In this paper, we present the preliminary results in the application of artificial neural networks to the automatic generation of face animation models. The network "learns" the complex nonlinear mapping between position sensor readings on the face and the position of graphical polygons that best fit the real face. The following section describes the neural approach in some detail. The third and fourth sections discuss the data collection and preprocessing of 2D face silhouettes. The fifth section gives the results of the neural modeling of the face silhouettes, and the sixth section draws conclusions and discusses future work.

2. The Neural Solution

Artificial neural networks have been used in many applications to form highly nonlinear mappings between high dimensional feature spaces [11]. Although several different types of neural architectures can perform this function ([12], [13]), we focus in this paper on a class of networks called multilayer perceptrons. The architecture is characterized by a "feed forward" layered structure, where weighted signals from the outputs of neurodes (neuron node) in one layer flows into the inputs of the next layer. Such an architecture in shown in the central portion of Fig. 1.

The layers not directly accessible to the inputs or outputs are called hidden layers. The constituent neurodes are called hidden nodes. The network learns through a weight adjustment algorithm called, in this case, backpropagation ([14], [15], [16]). The training data consists of a set of example input/output pairs of vectors representing the mapping between the input space and the output space. A vector from the input set is presented to the input neurodes. The signals forward propagate through the network until they reach the output neurodes where they are compared to the associated output vector. Backpropagation learning performs an error reducing correction to the weights in the network for each presentation. The error is defined as the root-mean square difference between each neurodal output in the output layer and a vector of target output values, summed over all example input/output vector pairs in the training set. The training set is presented to the system multiple times in what are called presentation epochs, until the total error has reduced to an acceptable level. The number of hidden nodes determines the degree of generalization produced by the trained network when it is applied to a testing set of input/output vector pairs not previously presented to the network for learning. Too many hidden nodes and the system will memorize the data, too few hidden nodes and the system will poorly generalize previously unused pattern classifications.

Fig.1 illustrates how such a network learns a personal face model. A set of training data is collected from the person by a 3D scanner (such as the laser scanner of Cyberware) while they are talking and making a wide range of facial expressions. This data is then preprocessed (Sec 4.) and a small set of sampling points on the face is extracted. Distances from the sample points to nearby fixed points on a rigid reference surface are computed in software and are used as the input parameters to the model. The full set of face polygons is used as target values for the outputs of the model. This forms one example mapping in the training set composed of (input parameter, output polygons) pairs. Many such pairs are presented to the network in random order to constitute one training epoch. Many such epochs are performed to reduce the total error to an acceptable level. At this point, the network is trained and ready for production use. In production, the network weights are frozen and the network is used in forward propagation mode only.

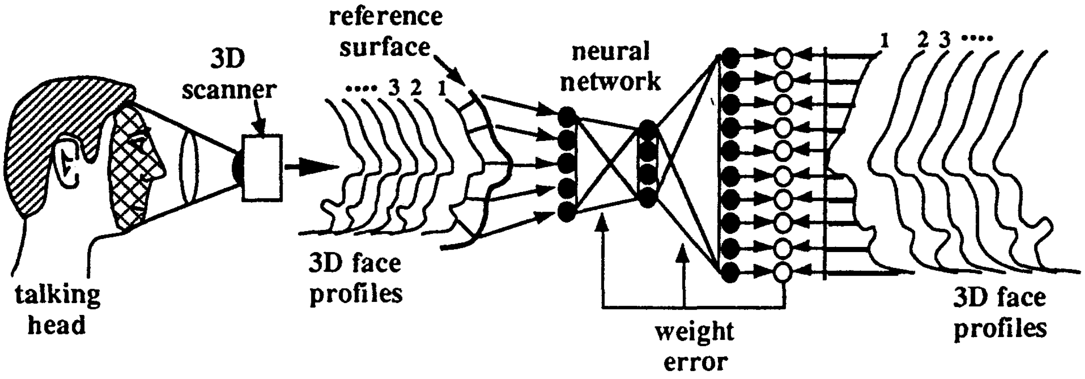

Figure 1. Diagram of training of network with face data.

Fig. 2 shows how the network is used in a telecommuting system. At startup time, after a telephone connection has been made, the face model is downloaded into the receiver's graphics engine. The transmitting individual dons a face mask that contains, in addition to the displays, a grid of gap sensors as seen in Fig. 3. The grid forms the physical counterpart of the reference surface used during training. Distance measurements are made in real time between the same selected sample points on the face used during training. As illustrated in Fig. 2, these distances are relayed to the inputs of the remote neural network over the telephone lines. At the other end, the network forward propagates the inputs to produce and display in real time an animated polygonal estimate of the original face. The gap sensors are not addressed in this paper, although either optical, acoustic, or electromagnetic technology may be used for their implementation.

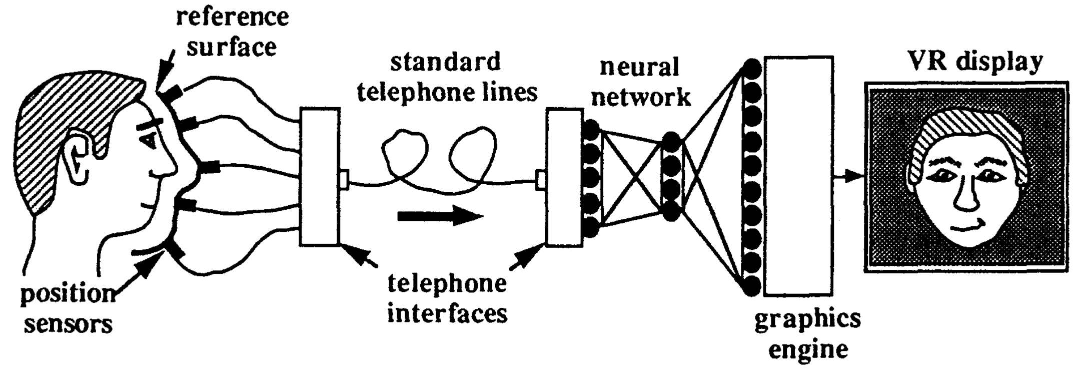

Figure 2. Diagram of neural network operational system

Several issues need to be resolved before this system can be implemented. First, the ability of a neural network to capture and generalize the expressions and conformations of facial expressions needs to be proved. Second, a procedure for the selection of the smallest set of sampling points must be devised. To begin addressing these two issues, we used a somewhat simpler set of face data in a prototype system.

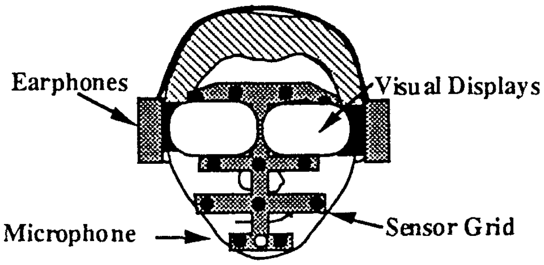

Figure 3. Example sensor points and a close up of Face Mask / Display

3. Face Silhouette Data Collection

To demonstrate the feasibility and to study the practical issues in the generation of neural face models, we generated models of 2D face silhouettes. Fig. 4 shows a diagram of the experimental setup for data collection. The subject stands with her shoulder against a wall near a convex corner. The room behind the corner is brightly illuminated, forming a high contrast scene. A black and white video camera mounted on a tripod and interfaced to a standard VHS video recorder is trained on the head of the subject as seen in Fig.4. The scale of the image is adjusted to leave a sharp baseline edge above and below the talking head. While holding as still as possible, the subject talks and makes facial expressions as the session is taped. No attempt has been made to control what is said or the expressions made.

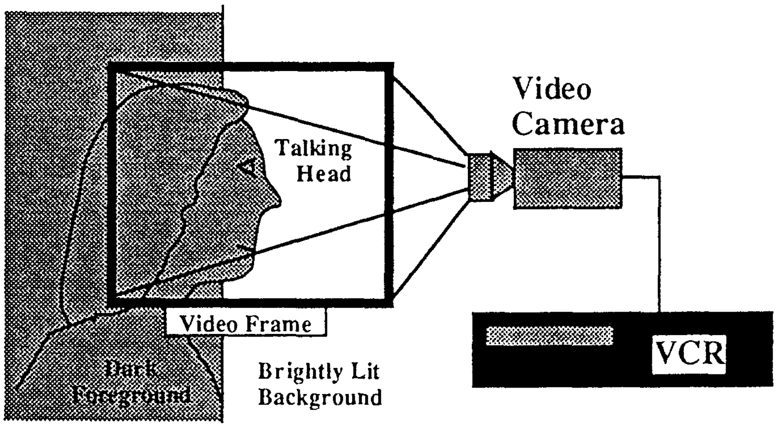

Figure 4. Configuration of high contract video silhouette collection system

For this pilot study, approximately five minutes of tape were produced. The tape was taken to a computer with a video frame grabber card for digitization. There, the tape was played into the computer and frames were grabbed as fast as possible. For this system, the grab rate was approximately once per second, leading to 299 face silhouettes. The eight-bit deep digitized 640x480 pixel image was converted into an edge list by a simple thresholding technique. Each row was searched from the right to find the column number where the intensity dropped to half its maximum value. A set of 480 edge column numbers was generated for each of the 299 faces. Following this, an average baseline edge was subtracted from each edge list and the result was stored for subsequent analysis.

4. Data Preprocessing

It is necessary to preprocess the data before presentation to the neural network for two reasons. First, glitchy or bad digitization scans must be removed. Second, the data must be transformed into a uniform location and scale because it is difficult for this type of network to learn "invariances" from a small set of data. After deglitching and rejecting bad edge lists from the face data, 229 sillhuettes remained.

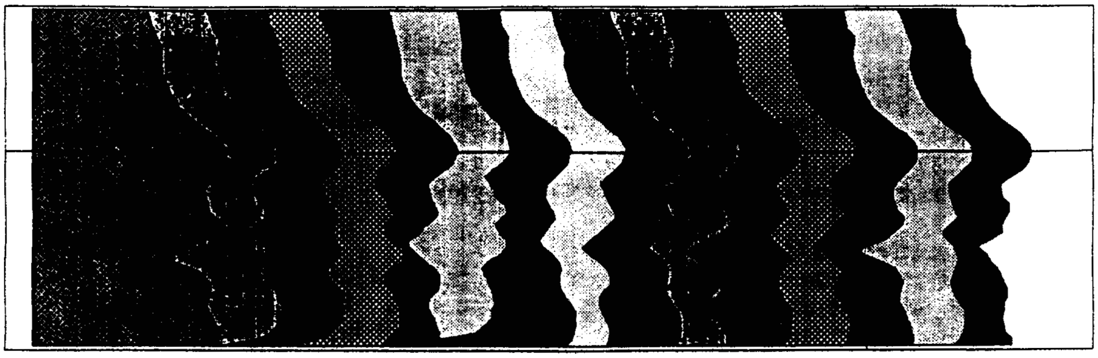

Figure 5. Processed, smoothed, and translated face edge profiles for a short sequence .

The transformation to a uniform position provided an interesting challenge. After perusing the data a number of times, we discovery the nose invariant -- the tip of the nose in this subject did not change shape significantly throughout the data set. We therefore adopted this point as an invariant, and translated all edge lists both in row and column number to a standard nose tip location. After translation, the edge lists were truncated at the top and bottom to include the largest number of rows common in all lists, reducing the number of rows from 480 to 240. A few examples from the sequence of processed data are shown in Fig. 5 after smoothing with a five point triangle convolution filter and normalization to the [0,1] interval. The mean and standard deviation for the final set of face edge profiles are given in Fig. 6.

Since the edge positions vary smoothly along the face profile, and there exist sizable linear regions in most face samples, we conclude that not all of the edges in the list are necessary. We therefore applied a thinning filter which amounted to an automatic piecewise linear segmentation of the edges based on a population variance test on the linear pieces. Basically, this procedure forms the largest linear pieces, or 1D polygons, that tessellate the edge profile without exceeding a maximum error over the entire data set. Similar algorithms exist for 2D polygons. In this case, we set the error to be 0.3 times the original quantization error in the raw edge data, reducing the number of edges from 240 to 39. These larger 1D polygons were used in the training of the neural model.

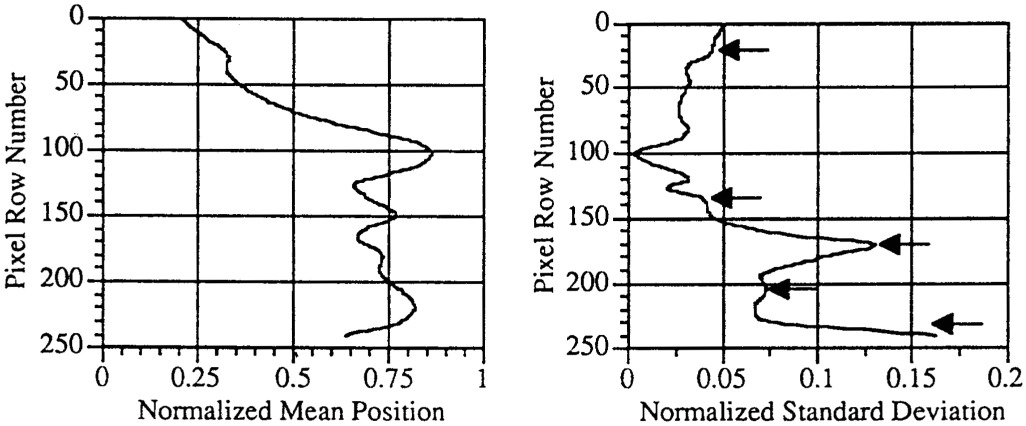

Figure 6. The mean and standard deviation face edge profile computed over the complete data set.

Before the network can be trained, the sample points must be selected. For this pilot study, the sample number and locations were manually picked based on the locations of the largest standard deviations plotted in Fig. 6 The arrows in the figure indicate their locations. More sophisticated optimization techniques based on maximizing regions of linear correlations have been designed, but were not necessary for this preliminary demonstration.

Trial	Hidden Units	Number of Epochs	% in Testing Set	RMS Error Training	RMS Error Testing
1	2	1000	25%	0.0406	0.0395
2	5	1000	25%	0.0296	0.0341
3	5	3000	50%	0.0284	0.0310
4	8	900	25%	0.0299	0.0272
5	10	1000	25%	0.0268	0.0287
6	15	2000	25%	0.0264	0.0307
7	20	175	25%	0.0379	0.0353

Table 1. Specific network parameters and fitting results for several network configurations.

5. Results of Neural Modeling

The multilayer perceptron used for this study had five input and 39 output neurodes. The number of hidden nodes was varied empirically. Training was performed with the vanilla backpropagation algorithm [17]. The learning rate was 0.05 and the momentum term was 0.01 with asynchronous weight updates. The weights were initialized with uniform random numbers between ±0.2. The data set was randomly divided into different training and testing subsets for each trial. During a trial, at intervals of 25 epochs, a validation test was performed with the testing subset. Training was terminated when either the validation error began to increase or a maximum epoch limit was reached.

The results of several trials are given in Table 1. Column 1 lists the number of hidden nodes, column 2 lists the maximum allowable number of epochs, column 3 lists the percentage of the data set used for testing validation, column 4 lists the final root-mean-square error summed over all 39 output values and all training samples, and column 5 lists the final root-mean-square error summed over all 39 output values and all testing samples.

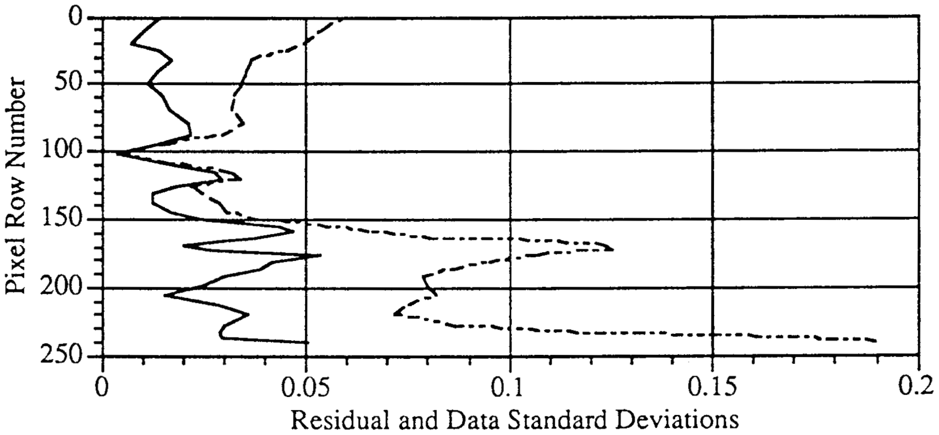

Figure 7. Testing data errors for Trail #5 in Table 1. Solid curve RMS residuals, dashed curve edge standard deviations.

To get a feeling for the goodness of these models, we plot in Fig. 7 the individual residual RMS errors for each output polygons along with the edge standard deviations for the testing data of Trial #5 in the table. The solid curve is the RMS error for the outputs while the dashed curve is the preprocessed data.

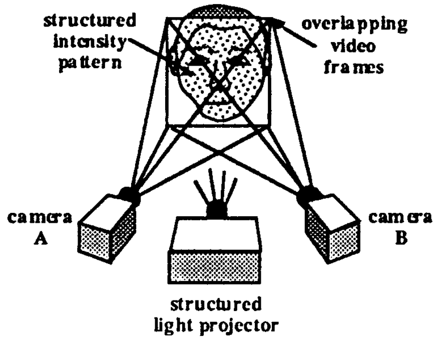

Figure 8. Configuration of stereo video face profiling using structured light

6. Discussion and Conclusions

The results given in Table 1 and Fig. 7 indicate that this simple neural model is to a large extent capturing the dynamics and deformations of this subject's facial expressions. In Fig. 7, the residuals are almost all significantly smaller than the variation found in the original edge data, showing that the network learned to generalize the nonlinear mapping between the sampled input measurements and the output face polygons. As a further demonstration of the performance of neural models, a video is being prepared for submission to the video proceedings.

Future work will involve larger 2D data sets, optimization of sampling points, experimentation with other neural architectures such as ARTMAP [12] and LAPART [13], and true 3D face data. Fig. 8 shows one possible method of collecting 3D face data currently under development using projected structured light and stereo metrology using video camera pairs [18].

Based on these experiments, we believe that neural networks have a potentially significant role to play in telecommunting and face model generation.

Acknowledgments

We would like to thank the subject talking head, Dr. Kathryn Ann Caudell, for facing up to this study and Drs. David Mizell and Henry Sowizral for many helpful comments and support.

References

1. J. M. Nilles, "Telecommuting and Organizational Decentralization", IEEE Trans. on Communications, Vol. 23, No. 10, pp. 1142-1147, 1975.

2. J. Fraser, "U.S. Telecommuting: Has Its Time Come", SRI International Technical Reprot #D91-157, 1991.

3. M. Quaid and B. Lagerberg, "Puget Sound Telecommuting Demonstration, Executive Summary", Washington State Energy Office Report #92-138, 1992.

4. M. Rose, "The Cocooning of America", Direct Marketing, pp.55-61, Feb. 1990.

5. F. I. Parke, "Computer Generated Animation of Faces", Proc. ACM Natil Conf., 1:451-457, 1972.

6. J. Kleiser, "A Fast, Efficient, Accurate Way to Represent the Human face", SIGGRAPH '89 Tutorial Notes: State of Art in Facial Animation, No 22, pp. 37-40, 1989.

7. M. Platt and N. I. Badler, "Animating Facial Expressions", Proc. SIGGRAPH '81, Computer Graphics, Vol. 15, No 3, pp. 245-252, 1981.

8. K. Waters, "A Muscle Model for Animating Three-Dimensional Facial Expressions", Proc. SIGGRAPH '87, Computer Graphics, Vol. 21, No 3, pp. 17-24, 1987.

9. B. deGaf, Notes on Facial Animation, SIGGRAPH '89 Tutorial Notes: State of Art in Facial Animation, No 22, pp10-11.

10. L. Williams, 'Performance Driven Facial Animation", Proc. SIGGRAPH '90, Computer Graphics, Vol. 24, No 3, pp.235-242.

11. R. D. Lipmann, "An Introduction to Computing with Neural Networks", IEEE ASSP, Vol. 4, No.22, 1987.

12. G. Carpenter, S. Grossberg, and J. Reynolds, "ARTMAP: Supervised Real-Time Learning and Classification of Non stationary Data by a Self-Organizing Neural Network", Neural Networks, Vol. 4, pp. 565-588, 1991.

13. J. Healy, T. P. Caudell, and S. D. G. Smith. "A Neural Architecture for Pattern Sequence Verification Through Inferencing", IEEE Transactions on Neural Networks, Vol. 4, No. 1, 1993.

14. E. Rummelhart and J. L. McClelland, Parallel Distributed Processing -- Explorations in the Microstructure of Cognition, Vol 1, Ch. 8, MIT Press Cambridge, Mass., 1986.

15. D. Parker, "Learning Logic", Invention report, S81-64, File 1, Office of Technology Licensing, Stanford University, 1982.

16. P. Werbos, "Beyond Regression: New Tools for Pridection and Analysis on the Behavioral Sciences", Ph.D. Dissertation, Harvard University, 1974.

17. K. Simpson, "Artificial Neural Systems -- Foundations, Paradigms, Applications, and Implementations", Pergamon Press, Ch. 5, p. 115, 1990.

18. S. K. Johnson, A. L. Janin, and T. P. Caudell, "See-thru Virtual reality Registration", Boeing Computer Services, Research and Technology Tech. Report #BCS-CS-ACS-92-006, 1992.